

**EXAMINATIONS – 2019**

**TRIMESTER 2**

**COMP 132 Term Test**  
**Programming for the Natural**  
**and Social Sciences**

**Time Allowed:** 50 minutes. If we start late, we will finish late.

**OPEN BOOK**

**Permitted materials:** All materials, calculators and dictionaries are permitted. Communication (e.g phone, chat, email, posting question/answers etc) is not allowed.

**Instructions:**

Before the test starts:

- Login to lab computer, create your file, rename your file to termtest.ipynb, and write your name and ID as comments
- Find the term test page by clicking the `Term Test` link on the green menu of our course home page. Open the data file on this page and you will need to copy and paste from this file.
- You can read the cover page of the test paper.
- You are NOT allowed to open the test paper to read any questions

Start:

- Open the test paper
- Save your file frequently to avoid data loss.

End:

- Save your code and do not modify it anymore.
- Submit it immediately to the right group using the link on the term test page.
- Hand in the hard copy of the test paper.

**Questions**

1. Using if [5]
2. Writing loops [5]
3. Using strings [8]
4. Data analysis using Numpy arrays [8]
5. Data visualisation using Matplotlib [8]
6. Using lists [16]

**Question 1. Using if****[5 marks]**

Write code to prompt the user to input a word and an integer number.

- If the word is more than 5 characters long and the number is bigger than 400, print "too long!";
- If the number is zero, print "Zero is not allowed";
- For all other cases, print out the word and the number.

**Question 2. Writing loops****[5 marks]**

Write a `for` loop to print the integer numbers from 1 to 20, and also add up all the numbers. Print the total at the end.

**Question 3. Using strings****[8 marks]**

Copy the Question 3 data (the string) in the `data.txt` file and use it in code to do the following:

- (a) **[3 marks]** Print the number of times that the letter "e" appears in the string.
- (b) **[5 marks]** Print the string with every letter "n" replaced by "\*".

**Question 4. Data analysis using Numpy arrays****[8 marks]**

Start your program by importing the Numpy library:

```
import numpy as np
```

Copy the Question 4 data from `data.txt`. Suppose the following data is the number of cars sold by two car dealers in each month of 2018 from January to December.

Dealer A: 36, 20, 40, 39, 47, 39, 11, 23, 27, 36, 33, 35.

Dealer B: 15, 20, 10, 30, 7, 40, 0, 2, 9, 8, 10, 20.

Write code to complete the tasks below:

- (a) **[2 marks]** Create two **Numpy arrays** containing the numbers.
- (b) **[3 marks]** Find and print the difference between the total numbers of cars sold by Dealer A and Dealer B in 2018.
- (c) **[3 marks]** A good month is one when a dealer sold more than 15 cars. Write code to print out the number of sales and the index for every good month of Dealer B. You may show them as two separate sequences of numbers.

**Question 5. Data visualisation using Matplotlib**

**[8 marks]**

Copy the Question 5 data from `data.txt`.

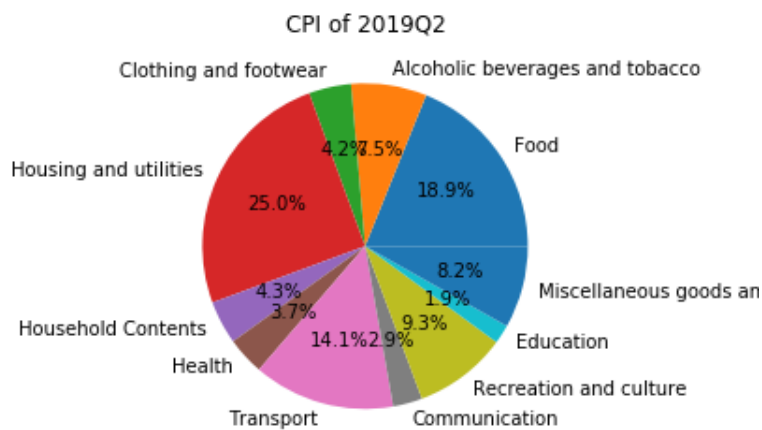
Set up your program as follows:

```
%matplotlib inline
import matplotlib.pyplot as plt
```

Write code (in one Jupyter cell) to visualise the data of New Zealand CPI (Consumers Price Index).

- (a) **[3 marks]** Plot the data for CPI using a pie chart;
- (b) **[3 marks]** Add the CPI values as percentages (e.g. 18.9%) inside the corresponding sector and display the categories outside the corresponding sector;
- (c) **[2 marks]** Add a title to the figure;

**HINT:** Your output should look like this:



**Question 6** is on the last page.

Spare page for your notes

**Question 6. Using lists**

**[16 marks]**

Copy the Question 6 data from `data.txt`.

The following data is the cash fares for Wellington trains based on the number of zones in the trip: 2.5, 4, 5, 5.5, 6.5, 8.5, 9.5, 10.5, 12, 13, 15, 16, 17.5, 19.

For example, passengers pay \$2.5 to travel within 1 zone, \$4 to travel between 2 adjacent zones and \$5.5 if the trip covers 4 zones, e.g. from zone 5 to zone 8.

(a) **[2 marks]** Use the `fare` list to print the fare for a trip of 5 zones.

The following lists are the stations of the train line KPL(Wellington-Waikanae) and the zone number (ID) that each station is in. For example, Wellington is in zone 1 and TakapuRoad is in zone 4.

```
station=['Wellington', 'TakapuRoad', 'Redwood', 'Tawa', 'Linden',
        'Kenepuru', 'Porirua', 'Paremata', 'Mana', 'Plimmerton',
        'PukeruaBay', 'Paekakariki', 'Paraparaumu', 'Waikanae']
```

```
zone=[1, 4, 4, 4, 4, 5, 5, 5, 6, 6, 7, 8, 9, 10]
```

(b) **[6 marks]** Write a loop to print each station and its zone. Your output should look like this:

---

```
Wellington is in zone 1
TakapuRoad is in zone 4
...
Waikanae is in zone 10
```

---

(c) **[8 marks]** Define a function with two parameters `start` and `dest` to calculate the fare to travel between a starting station and a destination station. You may use the lists `fare`, `station` and `zone`.

- Call your function to calculate the fare to travel from **Porirua** to **Waikanae**.
- Call your function to calculate the fare to travel from **Redwood** to **Tawa**.

**NOTE:** The fares can not be negative.

\*\*\*\*\*